242A Final Paper: Predicting Startup Success

Team Members: Veer Arora, Vincent Karpf, Paige Lyles, and Luis Schmitz

Motivation

It is common knowledge that about 80% of startups fail, including those with VC investments **(*Article 1*)** . The risk accrued is especially high for early-stage investments. What if we could reduce this risk and provide VC's a tool to inform their investments? A tool that predicts the success of a startup would allow VC funds to identify high-potential startups in a data-driven manner. For all purposes here, we will define "success" as having an evaluation of $100M or more. The aim of this project is to apply various Machine Learning techniques to predict if startups evaluated at their early-stage funding rounds, will achieve success or not. Various features that track pertinent aspects of startups will be used to classify whether a startup becomes a success or not.

Data

The data used in this project was sourced from PitchBook, an online database with information on private capital markets. This includes companies, deals, investors, funds, people, and investments. PitchBook does not provide a direct data download, necessitating data scraping. Easy Scraper, a Google Chrome extension was used to this end. Specifically, venture capital deal data was chosen by filtering on American VC backed companies founded between 2000-2018 with deal dates between 2000-2024. Companies 6 years or older were considered**.** The scraped data set contains 30,217 rows and 33 columns, with each row corresponding to a funding round. The dependent variable for the classification task was constructed using the post-money valuation of a company at their last funding round. Since we want to predict success at time of the first funding round we filtered the data set on the corresponding rows. A small sample of interesting features include: number of investors, deal size, industry related information, location, and CEO biographies. After cleaning and filtering we ended up with 10,525 viable data points (companies).

By finding patterns in the data structure, files A1-A3 cleaned the scraped data by ensuring data was in the correct columns and there were no rows being merged. Columns with significant levels of missing information, or those deemed redundant were dropped. The remaining columns were used in feature engineering **(*Feature Engineering Table*).** In the feature engineering stage (*Coding Files, A4*), 10 new numerical, binary, and categorical features were created using the available data and additional information. These features included identifying the number of investors, whether top VCs had invested (based on Pitchbook data), and whether the startup was part of an incubator. Additionally, information on the HQ location was considered, as being in a specific geographical region or a startup hub like the Bay Area or NYC could potentially influence success (*Article 3*). Another important factor is the age of the startup at the time of the deal, as it could reflect the company's or founders' growth ambitions and the investments into technology. As our later rudimentary feature analysis showed, these developed features were strong explanatory factors for predicting future startup success.

To extract valuable features from "free-text" columns such as "CEO Education" and "Keywords" we used NLP preprocessing. This includes removing punctuation, stopwords, and digits. In addition, we performed stemming and then constructed term features using TF-IDF vectorization to account for frequency of and relevance of terms. Moreover, we removed terms that appeared in more than 90% or

less than 1% of the rows in an attempt to exclude features with low predictive power. We also set a frequency threshold to limit the number of features extracted from each column, thereby reducing dimensionality. 550 features were produced by the NLP processing. The most prevalent features across the dataset are displayed in ***Figure 3***.

Categorical columns such as "Vertical" and "Primary Industry Sector" were converted to dummy variables. This was the last step to ensure that the data set can be inserted into the models. The final data set contains 874 features.

Modeling

After feature engineering, the next step is to build models and assess their performance (***Coding Files, A5***). VC funds aim at investing in a large portion of successful companies while trying to minimize the number of investments into companies that end up failing. Thus, we focused on two target metrics, precision and true positive rate, to maximize the portion of investments into successful companies out of all investments as well the portion of successful companies identified by the model out of all successful companies. First, we split the data set into a train (80%) and a test set (20%). Subsequently, all models were trained on the train set and optimized using the F1 score which weighs precision and TPR. It is also commonly used when dealing with class imbalance such as in this case. The data set is high-dimensional and therefore some features might not provide any predictive power, requiring feature selection and regularization. To find the best version of each model, we perform a grid search (GridSearchCV) over a grid of potentially effective values for the hyperparameters of each model. For each combination we evaluated the models using stratified K-fold cross validation to ensure equal class proportions in the folds.The modeling process is described in the following paragraphs.

Logistic Regression

To build the optimal logistic regression model we first performed feature selection using the elastic net penalty, which incorporates both L1 (Lasso) and L2 (Ridge) regularization. We first standardized all features. Then we used the following hyperparameters to find the optimal set of features: *L1_ratio* to control the scaling between the L1 and L2 penalties, the list of *alphas* along which to compute the models, and the number of features (*max_features*) selected from the model. Moreover, we only selected features with a feature importance score higher than the mean to reduce the number of irrelevant features that are fed into the model. Hyperparameters used to optimize the logistic regression model include *L1_ratio*, the *C* parameter (controls the inverse regularization strength), and the *class_weight* parameter. After looking at the shape of the ROC curve (***Figure 4***) and the thresholds' impact on precision and TPR we decided to apply a threshold of 0.5 in the final model.

CART Classification

Since tree-based models automatically identify influential variables through some splitting rule, no additional feature selection was performed. To avoid overfitting we focused on *ccp_alpha* to control cost complexity pruning as well as *min_samples_leaf* as a splitting criterion to limit tree depth.

Random Forest Classification

To find the best random forest model we iterate over key hyperparameters, including the number of estimators (*n_estimators*), maximum number of features (*max_features*), minimum samples per leaf node (*min_samples_leaf*), and *class_weight*. This allowed us to control model complexity in terms of number and depth of trees, ultimately reducing overfitting and improving generalization. We were also able to increase model robustness by controlling the variance between the individual decision trees in each random forest model.

Gradient Boosting Classification

The parameter grid of the gradient boosting model focuses on the following parameters: (1) *n_estimators* (number of trees) and, (2) *learning_rate* to control the tradeoff between the number of iterations and step size for convergence. Smaller learning rates help prevent overfitting by gradually improving predictions. (3) *max_depth* and *min_samples_leaf* to ensure shallow trees that generalize well, preventing overfitting to sparse data patterns. (4) *max_features* to restrict the feature space used at each split, improving efficiency and reducing variance of final predictions. This setup is significant because it balances computational efficiency, prevents overfitting, and tailors the gradient boosting model to imbalanced or sparse data scenarios such as this one.

Analysis

We want to compare all models with an appropriate baseline model. We have a dataset of startups which all received VC or angel funding. Because this represents the actual behaviour of the VC industry the baseline strategy is to invest in all companies. Both TPR and FPR are 100% as this model does not predict any true negatives and false positives. Precision is 8.9%. ***Table 1*** summarises the performance of all models on the test set along key metrics, including accuracy, true positive rate (TPR), false positive rate (FPR), precision, counts of true and false positives, and counts of investments. This provides the basis for a detailed assessment of the models' ability to classify the test data effectively and provide value to VC firms. We recall our target metrics, precision and TPR, and also look at the number of investments as VC firms have a certain budget to allocate. The results vary significantly across models and the "best" model might depend on VC fund size and the number of companies available for analysis. While the gradient boosting model achieves the highest precision (55.56%) it identifies only a small portion of successes (TPR of 12.7%) and only classifies 36 out of 1765 companies as successful. This would make it a good choice for very small VC firms with a lot of company data. The CART model has a TPR of almost 20% and would lead to 121 investments, however, precision decreases to 25.6% which is the lowest among all models. Both logistic regression and random forest outperform the CART model in terms of both target metrics. They achieve precision scores of 32.10% and 37.95%, TPRs of 49.68% and 40.13%, and investment counts of 243 and 166, respectively, making them the preferred models for most VC funds. Note that the Logistic Regression model has the capability to move the threshold (***Figure 5***), meaning a firm can adjust how much risk they are willing to take. Since the profits of a few successful investments usually offset and ideally exceed the losses from many failed investments, a higher number of investments can lead to better performance. VC firms care primarily about profit and return on investment (ROI) while meeting their investment goals. Thus, the best model for a given fund

could be determined by simulating monetary performance on the test set using average profit and loss approximations. To simulate the fund performance discussed in the following section, we calculated these values based on our data sample (***Table 2***).

Since the primary goal of this project was providing a valuable prediction model to VC firms, generating insights on what features influence the probability of success wasn't prioritized. Nevertheless, we provided an overview of the most important model features in ***Figures 6 - 9***.

Test Classification Report					
	Logistic Regression	CART (Classification Tree)	Random Forest	Gradient Boosting	Baseline
Accuracy	0.8618	0.8776	0.8884	0.9133	0.0890
TPR	0.4968	0.1975	0.4013	0.1274	1
FPR	0.1026	0.0560	0.0641	0.0100	1
Precision	0.3210	0.2562	0.3795	0.5556	0.0890
# True Positives	78	31	63	20	157
# False Positives	165	90	103	16	1608
#Investments (TP+FP)	243	121	166	36	1765

Table 1: Model Performance Comparison

Impact

The machine learning models trained provide a step forward in data-driven venture capital. To compare its impact we simulated a VC fund's performance using these models comparing it to benchmark funds. As shown in ***Table 4*** of the appendix, the gradient boosting model achieved the highest internal rate of return (IRR) at 40%, outperforming well-established funds like Bessemer Venture Partners VIII (14%), EQT Ventures I (31%) or Lightspeed Venture Partners X (20%). However, this performance came with a more conservative investment strategy, resulting in only 36 investments with a total of $136.8M invested. On the other hand, models like logistic regression and random forest balance risk and reward more effectively for larger VC funds. Logistic regression led to 243 investments with an IRR of 32%, while random forest delivered an IRR of 35% across 166 investments.

We want to note that these models are performing very well compared to a few of the best funds in VC history and thus we want to acknowledge that the models are a bit "too good to be true", meaning that the models used may have feedback effects or biases (which we eliminated to our best) which may not lead to the same returns on future investments. However, when applied strategically, these models may aid VCs to maximize their ROI, identify high-potential startups earlier, improve decision-making accuracy, and optimize their portfolios. Despite these advantages, our models can bring potential biases through features like founder demographics or educational background. Thus, one should ensure that there are no ethical and inclusive issues when applying our models.

Appendix (Please download files from Links)

Files

1. Data Source: Pitchbook (Filter included)
2. Data Files (.csv)
    a. Raw: Raw scraped data
    b. AB1: Cleaned Data
    c. AB2: Feature Engineered Data
3. Feature Engineering Table (.xlsx file): Link
4. Coding Files (.ipynb file)
    a. A1: Data Extraction and Column Renaming
    b. A2: Data Cleaning and Combining
    c. A3: Feature Cleaning
    d. A4: Feature Engineering
    e. A5: Model Variation

Articles

1. Kizilkan, Katrin. "205 Startup Statistics: Trends, Rates, Funding, and Teams." *Flair Blog for HR Professionals*, flair Blog for HR Professionals, 12 Nov. 2024, flair.hr/en/blog/startup-statistics/.
2. Perlroth, Nicole. "Venture Capital Firms, Once Discreet, Learn the Promotional Game." *The New York Times*, The New York Times, 23 July 2012, www.nytimes.com/2012/07/23/business/venture-capital-firms-once-discreet-learn-the-promotional-game.html.
3. Startup Genome. "Silicon Valley Leads: Discover North America's Top Startup Ecosystems." *Startup Genome*, startupgenome.com/articles/silicon-valley-leads-discover-north-americas-top-startup-ecosystems. Accessed 18 Dec. 2024.
4. *"How Long Do Startups Take to Become Profitable?" RSS, www.ninetwothree.co/blog/how-long-do-startups-take-to-become-profitable. Accessed 18 Dec. 2024.*

Figures

Orignial Attributes

Attribute	Description	Use for Model
Year Founded	The year the company was established.	Yes
Keywords	Specific terms associated with the company's business activities, products, or services. (e.g. ag biotech or blood pressure monitor)	Yes
Vertical	A specific niche or specialized market that the company operates in, spanning multiple industries. (E.g. Digital Health, HealthTech)	Yes
CEO Biography	A summary of the Chief Executive Officer's professional background and experience.	Yes
CEO Education	A summary of the CEOs educational history.	Yes
Primary Industry Code	A standardized keyword for representing the main industry in which the company operates. (e.g. Application Software)	Yes
Business Status	The operational state of the company (e.g., Generating Revenue or Stealth).	Yes
Deal Type 2	An additional classification of the deal type (e.g. Series A, Series B).	Yes
Primary Industry Sector	A generalised sector or business model code (e.g. Consumer Products and Services (B2C) or Financial Services).	Yes

Figure 1

Engineered Features

Attribute	Reasoning	Use for Model
Numb of unique new Investors	Numerical information on number of new investors	Yes
Top100earlystageVC	Top VC (early-stage and sorted by performance - Pitchbook Data), do better VCs improve unicorn chance	Yes
Top100SeedVC	Top VC (seed and sorted by amount of investments - Pitchbook Data), do better VCs improve unicorn chance	Yes
Top100Accelerator	Accelerator (top 100 accelerator/incubator sorted by total investments - Pitchbook Data), do Accelerators or Incubators have an effect on unicorn chance	Yes
Year of Deal	Time Fixed Effects	Yes
Company Age Deal Date	Maturity of Company at Deal	Yes
Business Quarter at Deal	Capturing Seasonality Trends through Business Quaters	Yes
HQ Geographical Region	Feature Reduction of HQ Location and Regional Effects	Yes
HQ in a Startup Hub	State if HQ Location is in a Startup Hub (Source: startupgenome.com)	Yes
Deal No. > 1	There were earlier deals/ money raised (Not VC/ Angel Rounds, Family Investors etc.)	Yes

Figure 2

Most prevalent features in term matrices

Company Keywords

platform_key	1730.263301
manag_key	1580.645551
data_key	1280.521293
servic_key	1241.303268
technolog_key	1211.094079
product_key	1159.798325
softwar_key	1007.984952
system_key	960.025256
analyt_key	910.308906
devic_key	852.778659

CEO Education

univers_edu	2871.648050
bachelor_edu	1961.148708
scienc_edu	1955.510098
busi_edu	1420.955844
master_edu	1386.638668
degre_edu	1302.853896
bs_edu	1204.418022
art_edu	1064.041335
administr_edu	1054.258812
engin_edu	1035.987131

Figure 3

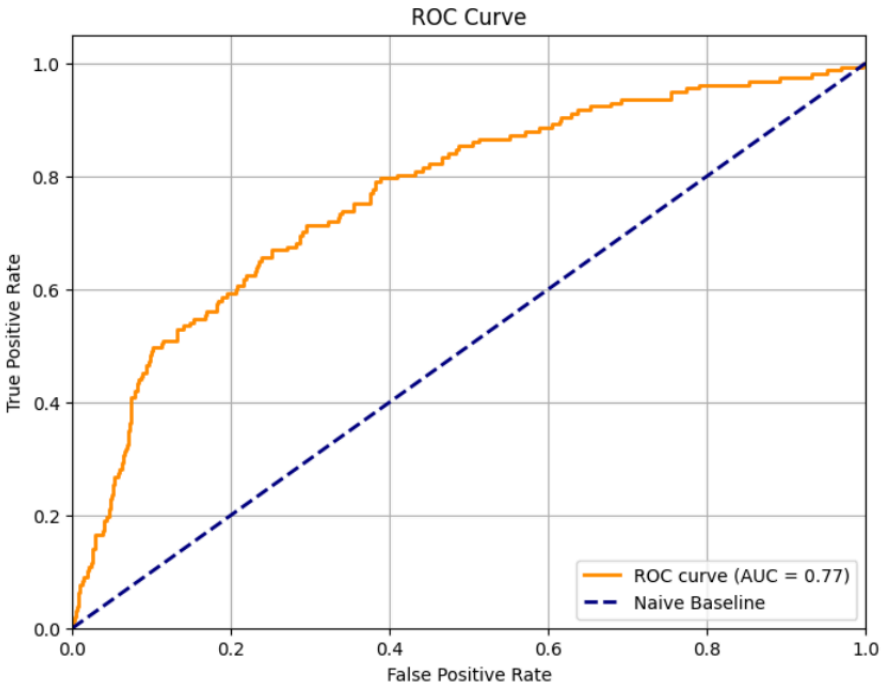

Figure 4 *(logistic regression model results –ROC Curve)*

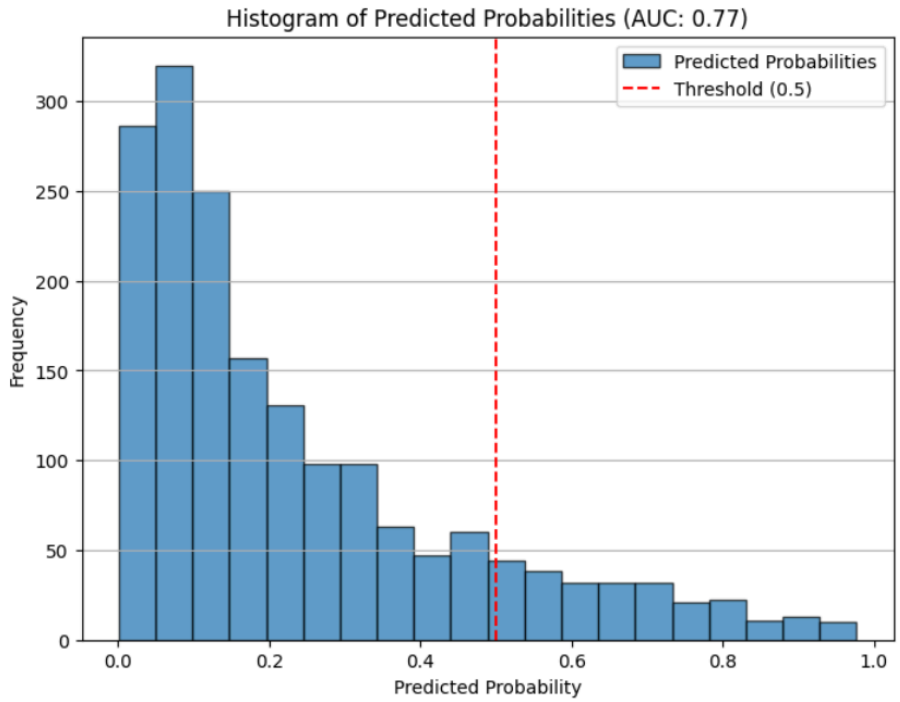

Figure 5 *(logistic regression model results – probability threshold)*

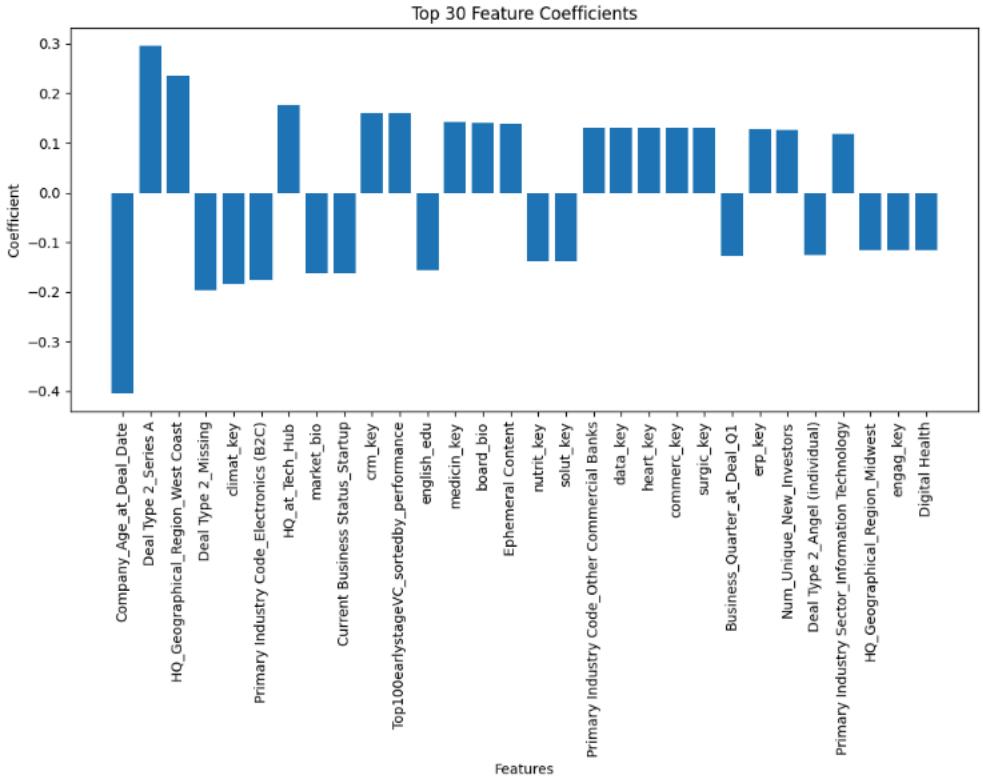

Figure 6 *(Logistic Regression model results - feature importance)*

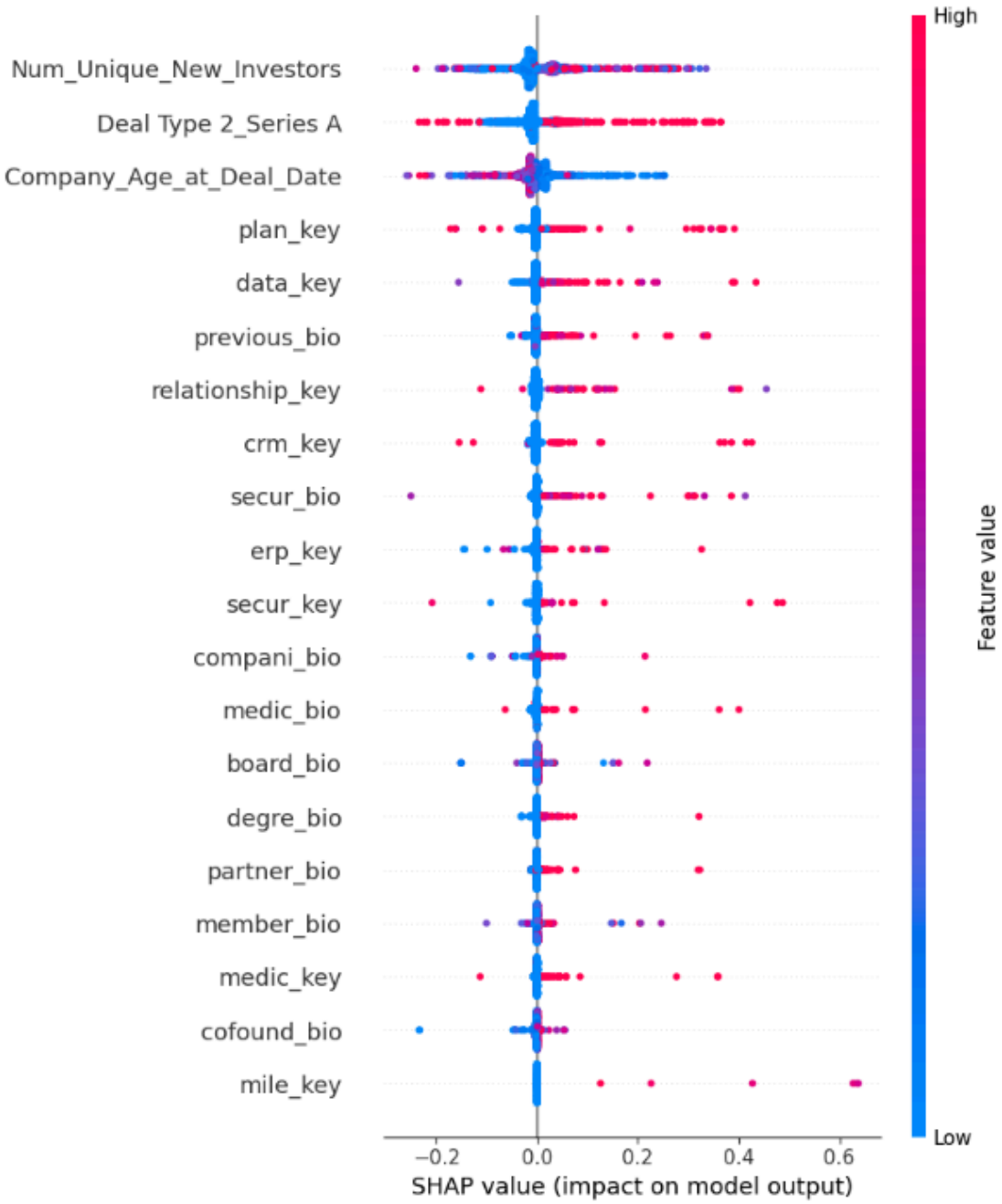

Figure 7 *(CART model results - feature importance)*

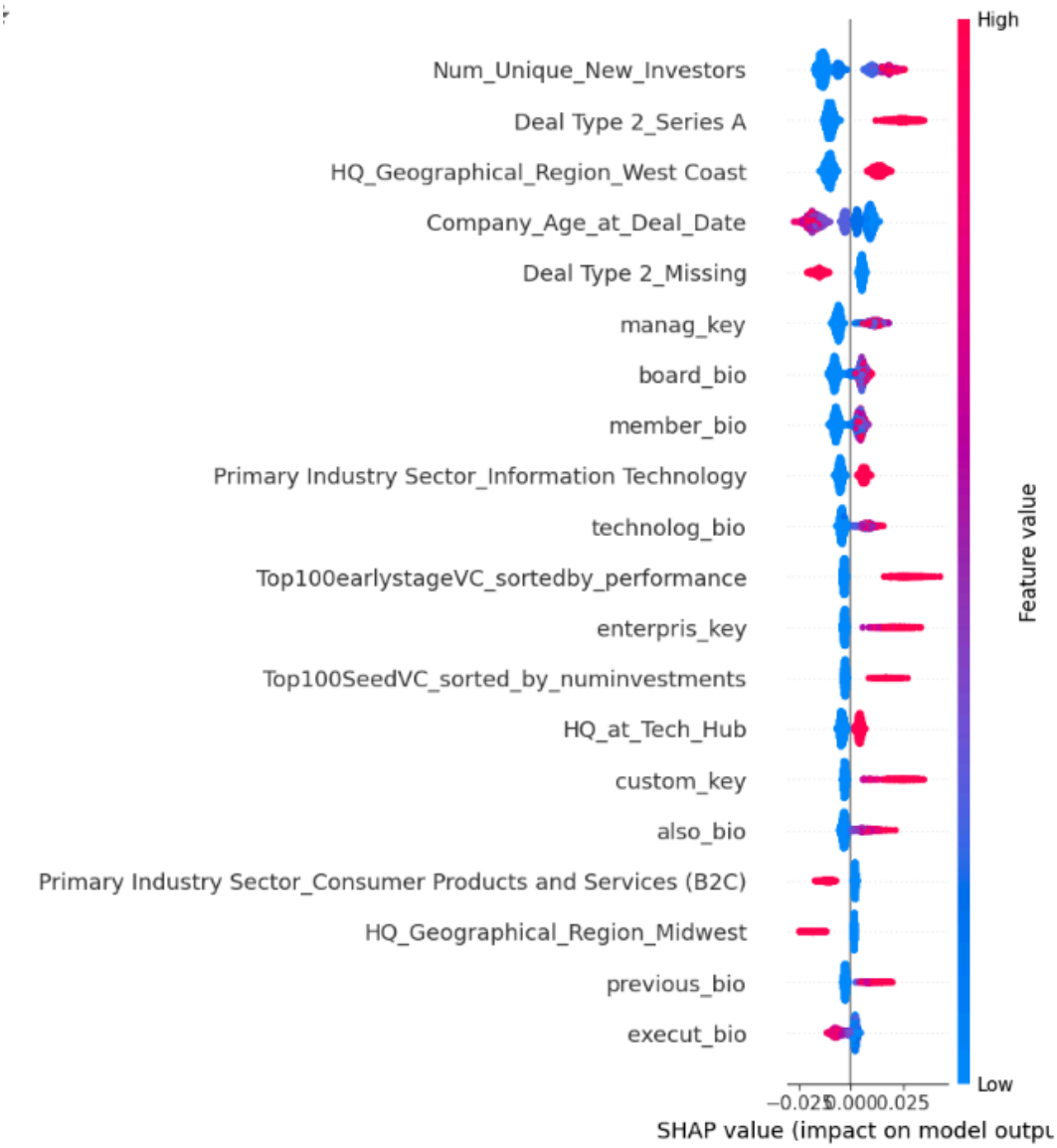

Figure 8 *(Random Forest model results - feature importance)*

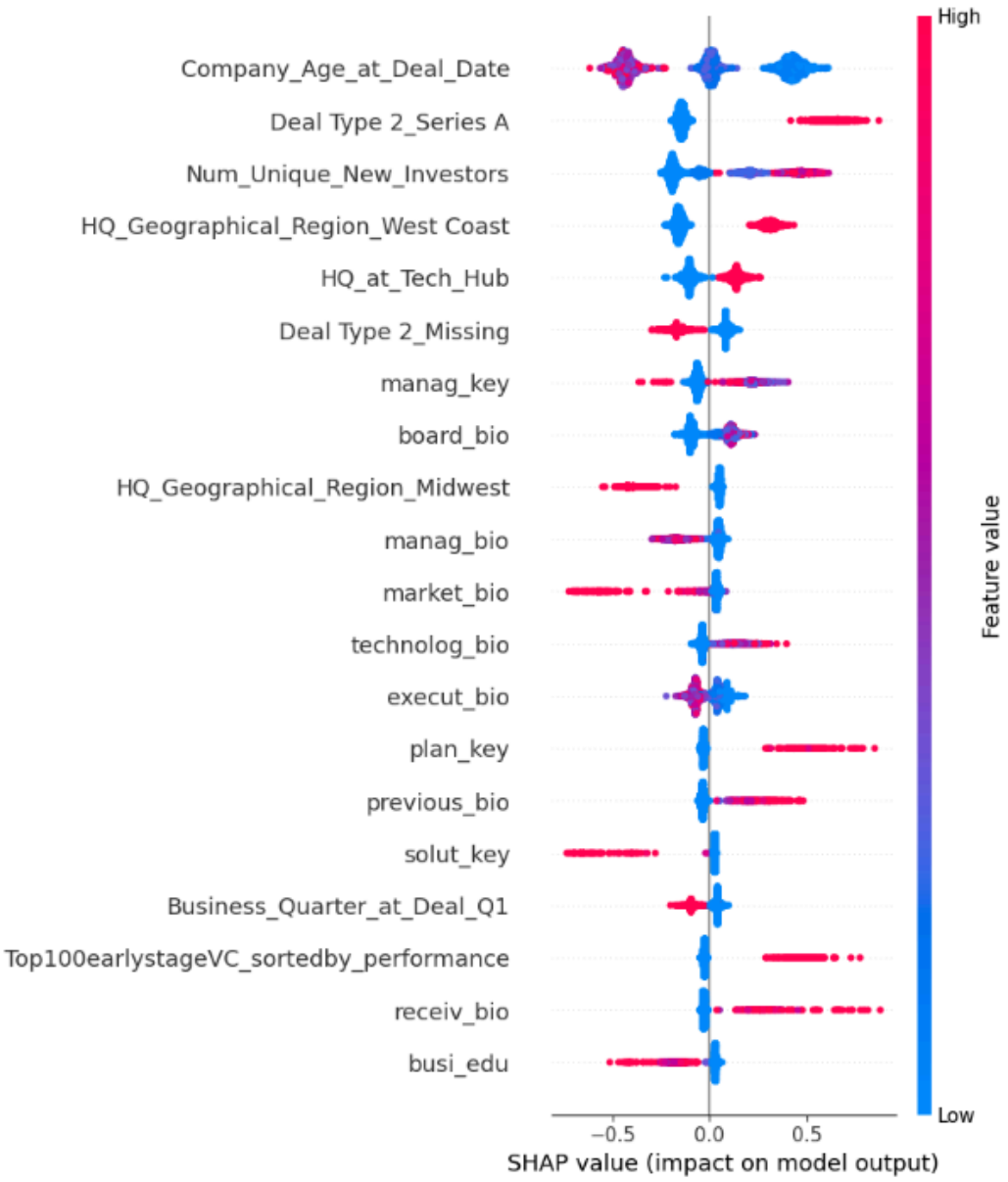

Figure 9 *(Gradient Boosting model results - feature importance)*

	Profit
count	618.000000
mean	201.816302
std	498.116102
min	0.000000
25%	37.888571
50%	68.515947
75%	166.342778
max	8717.180233

	Deal Size
count	7053.000000
mean	3.803973
std	14.521919
min	0.010000
25%	0.450000
50%	1.300000
75%	3.160000
max	654.000000

Figure 10 *(Statistics on the mean profit of a successful startup and also the mean deal size in the given data)*

Impact Calculations

The following calculations demonstrate the performance of our models if applied in a VC fund setting, assuming that all positive predictions meant an investment. The gain or loss per fund, as well as the return on investment (ROI) and internal rate of return (IRR) over a typical 10-year fund lifecycle, are computed based on average deal size and the average return of a successful investment. It is assumed that "unsuccessful" startups result in a total loss of the invested capital.

For comparison, data from PitchBook is used to benchmark against large, well-established funds that have achieved some of the most exceptional returns in VC history.

Assumptions (Empirical results from Data)	
Mean Gain	$ 201,800,000.00
Mean Loss/ Median Investment	$ 3,800,000.00

Table 2 *(Assumption on mean gain and loss of a startup investment in our sample)*

Model as a Fund

Model	TP	FP	Gain / Loss	Invested Capital	ROI	IRR (10years)	IRR (20 years)
Logistical Regression	78	165	$ 15,113,400,000	$ 923,400,000	1637%	32%	15%
CART	31	90	$ 5,913,800,000	$ 459,800,000	1286%	29%	14%
Random Forest	63	103	$ 12,322,000,000	$ 630,800,000	1953%	35%	16%
Gradien Boosting	20	16	$ 3,975,200,000	$ 136,800,000	2906%	40%	18%

Table 3 *(Modelling the different ML models as a VC fund)*

Impact Fund Name	Invesment Stage	IRR	Money Invested	Number of Investments	Average Investment
VC Funds					
Bessemer Venture Partners VIII	(more Late stage)	14%	$ 1,022,000,000	79	$ 12,936,709
EQT Ventures I	(Seed to Series B)	31%	$ 575,000,000	74	$ 7,770,270
Lightspeed Venture Partners X	(Seed to Series C)	20%	$ 637,000,000	92	$ 6,923,913
Model					
Logistical Regression	(Pre-Seed to Series A)	32%	$ 923,400,000	243	$ 3,800,000
CART	(Pre-Seed to Series A)	29%	$ 459,800,000	121	$ 3,800,000
Random Forest	(Pre-Seed to Series A)	35%	$ 630,800,000	166	$ 3,800,000
Gradien Boosting	(Pre-Seed to Series A)	40%	$ 136,800,000	36	$ 3,800,000

Table 4 *(Comparison of ML models and VC funds)*